

**IN THE UNITED STATES
PATENT AND TRADEMARK OFFICE**

APPLICANT: David Raymond Larsen
SERIAL NO: 09/534,201
FILING DATE: March 24, 2000
TITLE: Reconciling Combinations of Transactions
EXAMINER: James A. Kramer
GROUP ART UNIT: 3627
ATTY. DKT. NO.: 4760

MAIL STOP APPEAL BRIEF - PATENTS
COMMISSIONER FOR PATENTS
P.O. BOX 1450
ALEXANDRIA, VA 22313-1450

APPEAL BRIEF

Real Party in Interest

The subject application is owned by Intuit Inc. of Mountain View, California.

Related Appeals and Interferences

There are no known prior or pending appeals, interferences or judicial proceedings known to Appellant, Appellant's legal representative, or assignee which may be related to, directly affect or be directly affected by, or have a bearing on the Board's decision in the pending appeal.

Status of Claims

Claims 1-54 are pending and stand finally rejected.

Status of Amendments

Appellant has not amended the claims since the final rejection dated September 28, 2006.

Summary of Claimed Subject Matter

In compliance with 37 CFR § 41.37 (c)(1)(v), Applicant has referred in the summary below to the specification by page and line number and to the drawings by reference characters. Applicant notes, however, that the references below are by way of example only and are not exhaustive of the support found within the Application.

The claims recite a specific method for reconciling a first transaction in a first list with a combination of at least two transactions in a second list. (Appl. page 6, lines 2-4, page 9, line 18-page 10, line 3, page 12, lines 4-7, page 15, lines 5-9; FIG. 1, number 107, FIG. 2, Case 4760

numbers 201-217, FIG. 4; independent claims 1, 6, 21, 26, 40, 43). The claims also recite a specific method for reconciling a first combination of at least two transactions in a first list with a second combination of at least two transactions in a second list. (Appl. page 6, lines 2-4, page 9, line 18-page 10, line 3, page 12, lines 15-17, page 15, lines 5-9, page 16, lines 14-16, page 18, lines 4-10; FIG. 1, number 107, FIG. 2, numbers 201-217, FIG. 4; independent claims 14, 33, 49).

The claims further recite a specific system and method for matching a first value with a combination of at least two values in a list of values. (Appl. page 6, lines 11-17, page 15, lines 5-9, page 16, lines 14-16; FIG. 1, number 107, FIG. 2, numbers 201-217, FIG. 4; independent claims 15, 34, 50). In addition, the claims recite a specific system and method for matching a first combination of at least two values with a second combination of at least two values in a list of values. (Appl. page 6, lines 17-19, page 15, lines 5-9, page 16, lines 14-16, page 18, lines 4-10; FIG. 1, number 107, FIG. 2, numbers 201-217, FIG. 4; independent claims 20, 39, 54). The claims also recite specific steps of the recursive method, and further recite conditions and mechanisms for the recursive method to call itself. (Appl. page 6, lines 20-page 7, line 2, page 15, lines 5-9, page 16, lines 14-16, page 17, lines 11-21; FIG. 1, number 107, FIG. 2, , numbers 201-217; independent claim 20).

Grounds of Rejection to be Reviewed on Appeal

Whether claims 1-54 are unpatentable under 35 U.S.C. § 103(a) over U.S. Patent Application No. 5,134,564 to Dunn et al. in view of Kreminiec ("Reconciling Accounts the Automated Way")?

Argument

The rejection is improper because both Dunn and Kreminiec fail to teach or suggest the claimed subject matter. To establish *prima facie* obviousness, all claim limitations must be taught or suggested by the prior art. See MPEP §2143.03.

The independent claims 1, 6, 14, 15, 20, 21, 26, 33, 34, 39, 40, 43, 49, 50, and 54 generally recite methods, systems, and computer products for reconciling a transaction in a first list with a combination of at least two transactions in a second list, each transaction having a value. As further claimed, the method “determine[s] whether the value of the first transaction corresponds to a combination of the values of a subset of transactions in the second list.”

Dunn and Kreminiec fail to teach such steps. With regard to claims 1, 6 and 15, as the Examiner correctly acknowledges, Dunn does not discuss matching a transaction from a first list to a subset of transactions from a second list. With regard to claim 14, Dunn also does not discuss the matching of multiples or combinations (many-to-many). Dunn describes only one-to-one matching.

Claim 1

With regard to claim 1, Kreminiec has only one pertinent sentence: “this software allows matching multiple transactions to a summary transaction, thereby reducing the number of unmatched items that require research.” At most, Kreminiec discusses the software’s desired outcome, not any techniques to obtain the outcome. There is no description whatsoever of particular steps, methodology, or architecture for achieving these

goals. Specifically, there is no discussion of a system or method for reconciling a first transaction in a first list with a combination of at least two transactions in a second list.

Moreover, Kreminiec does not provide screenshots, code or any sort of documentation as to what the software could or could not accomplish or how any such operations were performed. Moreover, it is not clear where the system was in its development at the time the article was published. The article indicates that the system was still in the process of being developed: “In the future, we intend to expand the capabilities of this system for maintaining and controlling various suspense [*sic*] accounts.” Kreminiec simply does not disclose what the system could do or how it operated.

To establish *prima facie* obviousness of a claimed invention, all claim limitations must be taught or suggested by the prior art. See MPEP §2143.03. The statement that “the Examiner can think of only two ways to perform one-to-many matching” and the attendant illustration on page 8 of the September 28, 2006 Office Action, are simply not a proper prior art reference or common knowledge, as discussed in MPEP 2144.03:

It would not be appropriate for the examiner to take official notice of facts without citing a prior art reference where the facts asserted to be well known are not capable of instant and unquestionable demonstration as being well-known. For example, assertions of technical facts in the areas of esoteric technology or specific knowledge of the prior art must always be supported by citation to some reference work recognized as standard in the pertinent art.

As such, the Examiner’s statement that “search/matching algorithms are old and well-known in the art and have been explained and discussed for [*sic*] in text books as one of the fundamental coding principles” cannot be used as evidence of obviousness. If the “search/matching algorithms” were in fact “old and well-known” and had been discussed “in

text books”, the Examiner certainly should have been able to find *one* text book or reference that does precisely that. Although this application has been pending for nearly six years, no such text book has been cited as a reference, and the cited references simply do not disclose the limitations set forth in the claims.

Claim 6

In addition to the discussion above with regard to claim 1, claim 6 contains a detailed description of how the method reconciles one-to-many transactions. For example, the Examiner has not shown where the references disclose the following elements recited in claim 6: responsive to the value of the first transaction corresponding to the value of a transaction in the second list, indicating a match between the first transaction and the transaction having the corresponding value; and responsive to the value of the first transaction not corresponding to the value of a transaction in the second list: determining whether the value of the first transaction corresponds to a combination of the values of a subset of transactions in the second list. Kreminiec simply does not disclose any techniques to obtain its desired outcome.

Claim 15

In addition to the discussion above regarding claim 1, the Examiner has not shown how the references disclose the following element of claim 15: performing a submethod, using a first input parameter including the first value and a second input parameter including the second list of values, to determine whether the first value corresponds to a combination of values from the second list. In addition to the deficiencies in the references discussed above as to claim 1, the references simply do not disclose any such input parameters.

Claim 14

In discussing the Kreminc reference, the Examiner states that the article teaches “one-to-many and many-to-many searching/matching algorithms.” On the contrary, Kreminc fails to provide any such teaching. Kreminc simply does not disclose how such matching is or could be carried out.

Claim 14 recites specific steps and elements for performing the reconciliation, many of which are not disclosed by Kreminc. For example, claim 14 recites such specific steps as “combining the obtained transactions to generate first value”, “determining whether the first value corresponds to a combination of the values of a subset of transactions in the second list”, and “responsive to the first value corresponding to the combination of values, indicating a match between the first combination and the subset of transactions.” None of these specific steps and limitations is taught by Kreminc. Rather, Kreminc discloses little more than the desired goal of performing many-to-many item matching, without any indication of how such a goal is accomplished or what steps were performed by the software at the time the article was written.

Thus, there is no hint or suggestion of specific steps, methodology, or architecture for performing the functionality to achieve those goals. In particular, there is no hint or suggestion of a system or method for reconciling a first transaction in a first list with a combination of at least two transactions in a second list, as claimed herein. Furthermore, the references fail to teach the specific steps recited in the claims.

Dependent Claims Reciting Recursive Methods (e.g., claims 11, 12, 16, 17, 31, 32, 35, 36, 46, 47, 48, 51, and 52)

The dependent claims also recite additional features and limitations that are not taught or suggested by Dunn or Kreminec. The claims that are dependent on claims 1, 6, and 15 specify how to perform the method in great detail.

For example, claims 11, 12, 16, 17, 31, 32, 35, 36, 46, 47, 48, 51, and 52 specifically recite recursive methods, and further recite particular steps for implementing and performing these recursive methods (including conditions for a submethod to call itself). There are any number of methods that can be used for performing one-to-many reconciliation, including some that may involve recursive methodologies and some that may not. These methods are simply not shown or discussed in Dunn or Kreminec.

Implications of Applicant's Prior 131 Declaration

Lastly, the Examiner made a conclusion, based on the fact that Applicant had previously filed a declaration under 37 CFR 1.131 to swear behind the Checkfree reference (previously cited in the Office Action dated August 15, 2005), that any argument against the Examiner's assumptions of "common knowledge" with regard to what the Kreminec reference shows (with regard to anticipation under § 103) is also an argument against the sufficiency of the showing set forth in Applicant's 1.131 declaration submitted previously as to the Checkfree reference (with regard to anticipation under § 102). This is not correct for several reasons.

As stated in MPEP 715.02, "a 37 CFR 1.131 affidavit is not insufficient merely because it does not show the identical disclosure of the reference(s) or the identical subject matter involved in the activity relied upon. If the affidavit contains facts showing a completion of the invention commensurate with the extent of the invention as claimed is

shown in the reference or activity, the affidavit or declaration is sufficient, whether or not it is a showing of the identical disclosure of the reference or the identical subject matter involved in the activity.” Thus, the determination of whether a 1.131 declaration is sufficient depends on the reference it is swearing behind. First, Applicant used the 1.131 declaration to swear behind the Checkfree reference. The Examiner correctly determined that the declaration was sufficient to swear behind Checkfree, as the declaration contains facts commensurate with the extent of the description set forth in Checkfree.

Now, in examining whether Dunn and Kreminiec are sufficient to render the invention obvious under 35 USC § 103(a), a separate determination takes place. As the MPEP makes clear, the determination of sufficiency of a 1.131 declaration depends on the showing made in the reference the Applicant is swearing behind. The determination of whether a different reference renders an invention unpatentable under 35 USC § 103(a) is not the same thing. **Applicant is not swearing behind Dunn and Kreminiec.** As such, Applicant’s prior declaration under Rule 1.131 to swear behind Checkfree (with regard to anticipation under §

102) is simply not pertinent to the determination whether Dunn and Kreminiec render the invention obvious under 35 USC § 103(a).¹

Examiner's Omnibus Rejection of Claims 1-54

Applicant also notes that the Examiner has rejected all of the Applicant's 54 claims without providing any explanation as to how the cited art applies to the elements of these claims. Because of the many distinct features of these claims, the Examiner's omnibus rejection is improper.² See M.P.E.P § 707.07 (d) ("A plurality of claims should never be grouped together in a common rejection, unless that rejection is equally applicable to all claims in the group.")

As described above, neither Dunn nor Kreminiec provide any teaching that anticipates the specific limitations recited in the claims of the present application. Therefore, it is respectfully requested that the final rejections of claims 1-54 be withdrawn.

¹ Applicant notes that even the standard for swearing behind under 35 USC § **102(a)** is different from the standard of swearing behind under 35 USC § **103(a)**. See 1-3 CHISUM ON PATENTS § 3.08 (2006) (citing *In re Tanczn*, 347 F.2d 830, 146 USPQ 298 (CCPA 1965)) ("When a rejection under 35 U.S.C. § 102(a) is involved, we have held that 'under the law all the applicant can be required to show is priority with respect to so much of the claimed invention as the reference happens to show.'...A different situation may prevail when the rejection is based upon 35 U.S.C. § 103. In such a case the purpose of an affidavit is to establish that the claimed invention was made by the applicant before the effective date of a reference relied upon to that the invention was obvious.") Applicant reiterates, however, that it is not swearing behind either Dunn or Kreminiec.

² See, for example, claims 11 and 12, which recite various elements of a recursive method.

Summary

For the foregoing reasons, Appellant believes that the Examiner's rejection of claims 1-54 was erroneous, and reversal of the decision is respectfully requested.

Respectfully submitted,
David Raymond Larsen

Dated: January 22, 2007

By: /Brenda M. Simon/

Brenda M. Simon, Reg. No. 48,449
Attorney for Applicant
Fenwick & West LLP
801 California Street
Mountain View, CA 94041
Tel.: (650) 335-7198
Fax: (650) 938-5200

Claims Appendix

The following are the claims involved in the appeal:

1. A computer-implemented method for reconciling a first transaction in a first list with a combination of at least two transactions in a second list, each transaction having a value, the method comprising:
 - obtaining the first transaction;
 - obtaining the second list of transactions;
 - determining whether the value of the first transaction corresponds to a
 - combination of the values of a subset of transactions in the second list; and
 - responsive to the value corresponding to the combination of values, indicating a
 - match between the first transaction and the subset of transactions.
2. The method of claim 1, wherein each transaction comprises one selected from the group consisting of an investment transaction, a financial transaction, and an accounting transaction.
3. The method of claim 1, wherein determining whether the value of the first transaction corresponds to a combination of the values of a subset of transactions in the second list comprises determining whether the value of the first transaction corresponds to a sum of the values of a subset of transactions in the second list.

4. The method of claim 1, wherein at least one of the steps of obtaining the first transaction and obtaining the second list comprises downloading transactions from a remote server.

5. The method of claim 1, wherein at least one of the steps of obtaining the first list and obtaining the second list comprises retrieving transactions from a storage device.

6. A computer-implemented method for reconciling a first transaction in a first list with a combination of at least two transactions in a second list, each transaction having a value, the method comprising:

obtaining the first transaction;

obtaining the second list of transactions;

determining whether the value of the first transaction corresponds to a value of a transaction in the second list;

responsive to the value of the first transaction corresponding to the value of a transaction in the second list, indicating a match between the first transaction and the transaction having the corresponding value; and

responsive to the value of the first transaction not corresponding to the value of a transaction in the second list:

determining whether the value of the first transaction corresponds to a combination of the values of a subset of transactions in the second list; and
responsive to the value corresponding to the combination of values, indicating a match between the first transaction and the subset of transactions.

7. The method of claim 6, wherein each transaction has a date, and wherein obtaining the second list comprises obtaining a list of transactions having dates identical to the date of the first transaction.

8. The method of claim 6, wherein each transaction has a date, and wherein obtaining the second list comprises obtaining a list of transactions having dates within a specified time period of the date of the first transaction.

9. The method of claim 8, further comprising, responsive to the value of the first transaction not corresponding to a combination of the values of a subset of transactions in the second list:

modifying the specified time period; and

repeating the steps of obtaining the second list, determining whether the value of

the first transaction corresponds to a combination of the values of a subset of transactions in the second list, and, responsive to the value corresponding to the combination of values, indicating a match between the first transaction and the subset of transactions.

10. The method of claim 1, wherein determining whether the value of the first transaction corresponds to a combination of the values of a subset of transactions in the second list comprises performing a recursive submethod using a first input parameter including the value of the first transaction and a second input parameter including the set of transactions in the second list.

11. The method of claim 10, wherein performing the recursive submethod comprises:

responsive to one of the values of a transaction in the second input parameter
equaling the first input parameter, returning a transaction list including the
transaction having the equal value;

responsive to none of the values of transactions in the second input parameter
equaling the first input parameter, and the second parameter containing
only one transaction, returning an indicator that no match was found;

responsive to none of the values of transactions in the second input parameter
equaling the first input parameter, and the second parameter containing
more than one transaction, performing the recursive submethod using a
modified first input parameter and a modified second input parameter, the
modified second input parameter omitting a selected transaction and the
modified first input parameter being obtained by subtracting the value of
the selected transaction from the first input parameter.

12. The method of claim 10, wherein performing the recursive submethod comprises:

responsive to one of the values of a transaction in the second input parameter
equaling the first input parameter, returning a transaction list including the
transaction having the equal value;

responsive to none of the values of transactions in the second input parameter
equaling the first input parameter, and the second parameter containing
only one transaction, returning an indicator that no match was found;
responsive to none of the values of transactions in the second input parameter
equaling the first input parameter, and the second parameter containing
more than one transaction, performing the steps of:

- a) selecting a transaction in the second input parameter;
- b) subtracting the value of the selected transaction from the first input
parameter to obtain a modified first input parameter;
- c) generating a modified set of transactions including all transactions in the
second input parameter except the selected transaction;
- d) performing the recursive submethod using a first input parameter including
the modified first input parameter and a second input parameter
including the modified set of transactions;
- e) responsive to the recursive submethod returning a transaction list, adding
the selected transaction to the returned list to generate a modified
transaction list, and returning the modified transaction list;
- f) responsive to the recursive submethod returning an indicator that no match
was found, performing the steps of:

responsive to any transactions remaining in the second
input parameter, repeating steps a) through f); and
responsive to no transactions remaining in the second input
parameter, returning an indicator that no match was
found.

13. The method of claim 1, further comprising repeating the obtaining,
determining, and indicating steps for a second transaction in the first list.

14. A computer-implemented method for reconciling a first combination of at least two transactions in a first list with a second combination of at least two transactions in a second list, each transaction having a value, the method comprising:

- obtaining each transaction in the first combination;
- combining the obtained transactions to generate a first value;
- obtaining the second list of transactions;
- determining whether the first value corresponds to a combination of the values of a subset of transactions in the second list; and
- responsive to the first value corresponding to the combination of values, indicating a match between the first combination and the subset of transactions.

15. A computer-implemented method for matching a first value with a combination of at least two values in a list of values, the method comprising:

- obtaining the first value;
- obtaining the second list of values;
- performing a submethod, using a first input parameter including the first value and a second input parameter including the second list of values, to determine whether the first value corresponds to a combination of values from the second list; and
- responsive to the first value corresponding to the combination of values, indicating a match for the first value.

16. The method of claim 15, wherein the submethod is recursive, and wherein performing the recursive submethod comprises:

responsive to one of the values in the second input parameter equaling the first input parameter, returning a value list including the equal value;

responsive to none of the values in the second input parameter equaling the first input parameter, and the second parameter containing only one value, returning an indicator that no match was found;

responsive to none of the values in the second input parameter equaling the first input parameter, and the second parameter containing more than one value, performing the recursive submethod using a modified first input parameter and a modified second input parameter, the modified second input parameter omitting a selected value and the modified first input parameter being obtained by subtracting the selected value from the first input parameter.

17. The method of claim 15, wherein the submethod is recursive, and wherein performing the recursive submethod comprises:

responsive to one of the values in the second input parameter equaling the first input parameter, returning a value list including the equal value;

responsive to none of the values in the second input parameter equaling the first input parameter, and the second parameter containing only one value, returning an indicator that no match was found;

responsive to none of the values in the second input parameter equaling the first input parameter, and the second parameter containing more than one value, performing the steps of:

- a) selecting a value in the second input parameter;
- b) subtracting the selected value from the first input parameter to obtain a modified first input parameter;
- c) generating a modified value list including all values in the second input parameter except the selected value;
- d) performing the recursive submethod using a first input parameter including the modified first input parameter and a second input parameter including the modified value list;
- e) responsive to the recursive submethod returning a value list, adding the selected value to the returned list to generate a modified value list, and returning the modified value list;
- f) responsive to the recursive submethod returning an indicator that no match was found, performing the steps of:

responsive to any values remaining in the second input parameter, repeating steps a) through f); and

responsive to no values remaining in the second input parameter, returning an indicator that no match was found.

18. The method of claim 15, wherein each value is associated with a transaction.

19. The method of claim 15, wherein the submethod determines whether the first value corresponds to a combination of values from the second list.

20. A computer-implemented method for matching a first combination of at least two values with a second combination of at least two values in a list of values, the method comprising:

- obtaining each value in the first combination;
- combining the obtained values to generate a first combined value;
- obtaining the second list of values;
- performing a recursive submethod, using a first input parameter including the first combined value and a second input parameter including the second list of values, to determine whether the first combined value corresponds to a second combination of values from the second list; and
- responsive to the first combined value corresponding to the second combination of values, indicating a match for each value in the first combination.

21. A computer program product comprising a computer-usable medium having computer-readable code embodied therein for reconciling a first transaction in a first list with a combination of at least two transactions in a second list, each transaction having a value, comprising:

- computer-readable program code devices configured to cause a computer to obtain the first transaction;
- computer-readable program code devices configured to cause a computer to obtain the second list of transactions;

computer-readable program code devices configured to cause a computer to determine whether the value of the first transaction corresponds to a combination of the values of a subset of transactions in the second list; and computer-readable program code devices configured to cause a computer to, responsive to the value corresponding to the combination of values, indicate a match between the first transaction and the subset of transactions.

22. The computer program product of claim 21, wherein each transaction comprises one selected from the group consisting of an investment transaction, a financial transaction, and an accounting transaction.

23. The computer program product of claim 21, wherein the computer-readable program code devices configured to cause a computer to determine whether the value of the first transaction corresponds to a combination of the values of a subset of transactions in the second list comprise computer-readable program code devices configured to cause a computer to determine whether the value of the first transaction corresponds to a sum of the values of a subset of transactions in the second list.

24. The computer program product of claim 21, wherein at least one of the computer-readable program code devices configured to cause a computer to obtain the first transaction and the computer-readable program code devices configured to cause a computer to obtain the second list comprises computer-readable program code devices configured to cause a computer to download transactions from a remote server.

25. The computer program product of claim 21, wherein at least one of the computer-readable program code devices configured to cause a computer to obtain the first transaction and the computer-readable program code devices configured to cause a computer to obtain the second list comprises computer-readable program code devices configured to cause a computer to retrieve transactions from a storage device.

26. A computer program product comprising a computer-usable medium having computer-readable code embodied therein for reconciling a first transaction in a first list with a combination of at least two transactions in a second list, each transaction having a value, comprising:

computer-readable program code devices configured to cause a computer to
obtain the first transaction;

computer-readable program code devices configured to cause a computer to
obtain the second list of transactions;

computer-readable program code devices configured to cause a computer to
determine whether the value of the first transaction corresponds to a value
of a transaction in the second list; and

computer-readable program code devices configured to cause a computer to,
responsive to the value of the first transaction corresponding to the value
of a transaction in the second list, indicate a match between the first
transaction and the transaction having the corresponding value; and

computer-readable program code devices configured to cause a computer to,
responsive to the value of the first transaction not corresponding to the
value of a transaction in the second list:

determine whether the value of the first transaction corresponds to a combination of
the values of a subset of transactions in the second list; and

computer-readable program code devices configured to cause a computer to,
responsive to the value of the first transaction corresponding to the
combination of values, indicate a match between the first transaction and
the subset of transactions.

27. The computer program product of claim 26, wherein each transaction has a date, and wherein the computer-readable program code devices configured to cause a computer to obtain the second list comprise computer-readable program code devices configured to cause a computer to obtain a list of transactions having dates identical to the date of the first transaction.

28. The computer program product of claim 26, wherein each transaction has a date, and wherein the computer-readable program code devices configured to cause a computer to obtain the second list comprise computer-readable program code devices configured to cause a computer to obtain a list of transactions having dates within a specified time period of the date of the first transaction.

29. The computer program product of claim 28, further comprising computer-readable program code devices configured to cause a computer to, responsive to the value of the first transaction not corresponding to a combination of the values of a subset of transactions in the second list:

modify the specified time period; and

repeat the steps of obtaining the second list, determine whether the value of the first transaction corresponds to a combination of the values of a subset of transactions in the second list, and, responsive to the value corresponding to the combination of values, indicate a match between the first transaction and the subset of transactions.

30. The computer program product of claim 21, wherein the computer-readable program code devices configured to cause a computer to determine whether the value of the first transaction corresponds to a combination of the values of a subset of transactions in the second list comprise computer-readable program code devices configured to cause a computer to perform a recursive submethod using a first input parameter including the value of the first transaction and a second input parameter including the set of transactions in the second list.

31. The computer program product of claim 30, wherein the computer-readable program code devices configured to cause a computer to perform the recursive submethod comprise computer-readable program code devices configured to cause a computer to:

responsive to one of the values of a transaction in the second input parameter
equaling the first input parameter, return a transaction list including the
transaction having the equal value;

responsive to none of the values of transactions in the second input parameter
equaling the first input parameter, and the second parameter containing
only one transaction, return an indicator that no match was found;

responsive to none of the values of transactions in the second input parameter
equaling the first input parameter, and the second parameter containing
more than one transaction, perform the recursive submethod using a
modified first input parameter and a modified second input parameter, the
modified second input parameter omitting a selected transaction and the
modified first input parameter being obtained by subtracting the value of
the selected transaction from the first input parameter.

32. The computer program product of claim 30, wherein the computer-readable
program code devices configured to cause a computer to perform the recursive submethod
comprise computer-readable program code devices configured to cause a computer to:

responsive to one of the values of a transaction in the second input parameter
equaling the first input parameter, return a transaction list including the
transaction having the equal value;

responsive to none of the values of transactions in the second input parameter
equaling the first input parameter, and the second parameter containing
only one transaction, return an indicator that no match was found;

responsive to none of the values of transactions in the second input parameter
equaling the first input parameter, and the second parameter containing
more than one transaction, perform the steps of:

- a) selecting a transaction in the second input parameter;
- b) subtracting the value of the selected transaction from the first input
parameter to obtain a modified first input parameter;
- c) generating a modified set of transactions including all transactions in the
second input parameter except the selected transaction;
- d) performing the recursive submethod using a first input parameter including
the modified first input parameter and a second input parameter
including the modified set of transactions;
- e) responsive to the recursive submethod returning a transaction list, adding
the selected transaction to the returned list to generate a modified
transaction list, and returning the modified transaction list;
- f) responsive to the recursive submethod returning an indicator that no match
was found, performing the steps of:

responsive to any transactions remaining in the second
input parameter, repeating steps a) through f); and
responsive to no transactions remaining in the second input
parameter, returning an indicator that no match was
found.

33. A computer program product comprising a computer-usable medium having
computer-readable code embodied therein for reconciling a first combination of at least two
transactions in a first list with a second combination of at least two transactions in a second
list, each transaction having a value, the computer program product comprising:
computer-readable program code devices configured to cause a computer to
obtain each transaction in the first combination;

computer-readable program code devices configured to cause a computer to
combine the obtained transactions to generate a first value;
computer-readable program code devices configured to cause a computer to
obtain the second list of transactions;
computer-readable program code devices configured to cause a computer to
determine whether the first value corresponds to a combination of the
values of a subset of transactions in the second list; and
computer-readable program code devices configured to cause a computer to,
responsive to the first value corresponding to the combination of values,
indicate a match between the first combination and the subset of
transactions.

34. A computer program product comprising a computer-usable medium having
computer-readable code embodied therein for matching a first value with a combination of at
least two values in a list of values, the computer program product comprising:

computer-readable program code devices configured to cause a computer to
obtain the first value;
computer-readable program code devices configured to cause a computer to
obtain the second list of values;
computer-readable program code devices configured to cause a computer to
perform a submethod, using a first input parameter including the first
value and a second input parameter including the second list of values, to

determine whether the first value corresponds to a combination of values from the second list; and
computer-readable program code devices configured to cause a computer to,
responsive to the first value corresponding to the combination of values,
indicate a match for the first value.

35. The computer program product of claim 34, wherein the submethod is recursive, and wherein the computer-readable program code devices configured to cause a computer to perform the recursive submethod comprise computer-readable program code devices configured to cause a computer to:

responsive to one of the values in the second input parameter equaling the first input parameter, return a value list including the equal value;
responsive to none of the values in the second input parameter equaling the first input parameter, and the second parameter containing only one value, return an indicator that no match was found;
responsive to none of the values in the second input parameter equaling the first input parameter, and the second parameter containing more than one value, perform the recursive submethod using a modified first input parameter and a modified second input parameter, the modified second input parameter omitting a selected value and the modified first input parameter being obtained by subtracting the selected value from the first input parameter.

36. The computer program product of claim 34, wherein the submethod is recursive, and wherein the computer-readable program code devices configured to cause a computer to perform the recursive submethod comprise computer-readable program code devices configured to cause a computer to:

responsive to one of the values in the second input parameter equaling the first input parameter, return a value list including the equal value;

responsive to none of the values in the second input parameter equaling the first input parameter, and the second parameter containing only one value, return an indicator that no match was found;

responsive to none of the values in the second input parameter equaling the first input parameter, and the second parameter containing more than one value, perform the steps of:

- a) selecting a value in the second input parameter;
- b) subtracting the selected value from the first input parameter to obtain a modified first input parameter;
- c) generating a modified value list including all values in the second input parameter except the selected value;
- d) performing the recursive submethod using a first input parameter including the modified first input parameter and a second input parameter including the modified value list;
- e) responsive to the recursive submethod returning a value list, adding the selected value to the returned list to generate a modified value list, and returning the modified value list;
- f) responsive to the recursive submethod returning an indicator that no match was found, performing the steps of:

responsive to any values remaining in the second input parameter, repeating steps a) through f); and responsive to no values remaining in the second input parameter, returning an indicator that no match was found.

37. The computer program product of claim 34, wherein each value is associated with a transaction.

38. The computer program product of claim 34, wherein the computer-readable program code devices configured to cause a computer to perform the submethod further comprise computer-readable program code devices configured to cause a computer to determine whether the first value corresponds to a combination of values from the second list.

39. A computer program product comprising a computer-usable medium having computer-readable code embodied therein for matching a first combination of at least two values with a second combination of at least two values in a list of values, the computer program product comprising:

computer-readable program code devices configured to cause a computer to obtain each value in the first combination;

computer-readable program code devices configured to cause a computer to combine the obtained values to generate a first combined value;

computer-readable program code devices configured to cause a computer to obtain the second list of values;

computer-readable program code devices configured to cause a computer to perform a recursive submethod, using a first input parameter including the first combined value and a second input parameter including the second list of values, to determine whether the first combined value corresponds to a second combination of values from the second list; and computer-readable program code devices configured to cause a computer to, responsive to the first combined value corresponding to the second combination of values, indicate a match for each value in the first combination.

40. A system for reconciling a first transaction in a first list with a combination of at least two transactions in a second list, each transaction having a value, the system comprising:

a first input device, for obtaining the first transaction;
a second input device, for obtaining the second list of transactions;
coupled to the first and second input devices, a memory for storing the first transaction and the second list;
coupled to the memory, a match determination module for determining whether the value of the first transaction corresponds to a combination of the values of a subset of transactions in the second list; and
coupled to the match determination module, a match indication module for, responsive to the value corresponding to the combination of values,

indicating a match between the first transaction and the subset of transactions.

41. The system of claim 40, wherein each transaction comprises one selected from the group consisting of an investment transaction, a personal financial transaction, and an accounting transaction.

42. The system of claim 40, wherein the match determination module determines whether the value of the first transaction corresponds to a sum of the values of a subset of transactions in the second list.

43. A system for reconciling a first transaction in a first list with a combination of at least two transactions in a second list, each transaction having a value, the system comprising:

a first input device, for obtaining the first transaction;

a second input device, for obtaining the second list of transactions;

coupled to the first and second input devices, a memory for storing the first transaction and the second list;

coupled to the memory, a transaction matching device for determining whether the value of the first transaction corresponds to a value of a transaction in the second list;

coupled to the transaction matching device, a match indication module for indicating, responsive to the value of the first transaction corresponding to the value of a

transaction in the second list, a match between the first transaction and the transaction having the corresponding value;

coupled to the memory, a match determination module for determining, responsive to the value of the first transaction not corresponding to the value of a transaction in the second list, whether the value of the first transaction corresponds to a combination of the values of a subset of transactions in the second list; and wherein the match indication module indicates, responsive to the value corresponding to the combination of values, a match between the first transaction and the subset of transactions.

44. The system of claim 43, wherein each transaction has a date, and wherein the second input device obtains a list of transactions having dates identical to the date of the first transaction.

45. The system of claim 43, wherein each transaction has a date, and wherein the second input device obtains a list of transactions having dates within a specified time period of the date of the first transaction.

46. The system of claim 40, wherein the match determination module performs a recursive submethod using a first input parameter including the value of the first transaction and a second input parameter including the set of transactions in the second list.

47. The system of claim 46, wherein the recursive submethod comprises:

responsive to one of the values of a transaction in the second input parameter
equaling the first input parameter, returning a transaction list including the
transaction having the equal value;

responsive to none of the values of transactions in the second input parameter
equaling the first input parameter, and the second parameter containing
only one transaction, returning an indicator that no match was found;

responsive to none of the values of transactions in the second input parameter
equaling the first input parameter, and the second parameter containing
more than one transaction, performing the recursive submethod using a
modified first input parameter and a modified second input parameter, the
modified second input parameter omitting a selected transaction and the
modified first input parameter being obtained by subtracting the value of
the selected transaction from the first input parameter.

48. The system of claim 46, wherein the recursive submethod comprises:
- responsive to one of the values of a transaction in the second input parameter
equaling the first input parameter, returning a transaction list including the
transaction having the equal value;
- responsive to none of the values of transactions in the second input parameter
equaling the first input parameter, and the second parameter containing
only one transaction, returning an indicator that no match was found;

responsive to none of the values of transactions in the second input parameter equaling the first input parameter, and the second parameter containing more than one transaction, performing the steps of:

- a) selecting a transaction in the second input parameter;
- b) subtracting the value of the selected transaction from the first input parameter to obtain a modified first input parameter;
- c) generating a modified set of transactions including all transactions in the second input parameter except the selected transaction;
- d) performing the recursive submethod using a first input parameter including the modified first input parameter and a second input parameter including the modified set of transactions;
- e) responsive to the recursive submethod returning a transaction list, adding the selected transaction to the returned list to generate a modified transaction list, and returning the modified transaction list;
- f) responsive to the recursive submethod returning an indicator that no match was found, performing the steps of:

responsive to any transactions remaining in the second input parameter, repeating steps a) through f); and
responsive to no transactions remaining in the second input parameter, returning an indicator that no match was found.

49. A system for reconciling a first combination of at least two transactions in a first list with a second combination of at least two transactions in a second list, each transaction having a value, the system comprising:

a first input device, for obtaining each transaction in the first combination;
coupled to the first input device, a combination module, for combining the obtained transactions to generate a first value;

a second input device, for obtaining the second list of transactions;
coupled to the combination module and the second input devices, a memory for
storing the first value and the second list;
coupled to the memory, a match determination module for determining whether
the first value corresponds to a combination of the values of a subset of
transactions in the second list; and
coupled to the match determination module, a match indication module for,
responsive to the first value corresponding to the combination of values,
indicating a match between the first combination and the subset of
transactions.

50. A system for matching a first value with a combination of at least two values in a
list of values, the system comprising:

a first input device, for obtaining the first value;
a second input device, for obtaining the second list of values;
coupled to the input devices, a memory for storing the first value and the second list;
coupled to the memory, a recursive function module, for performing a recursive
function, using a first input parameter including the first value and a second
input parameter including the second list of values, to determine whether the
first value corresponds to a combination of values from the second list; and
coupled to the recursive function module, a match indicator for, responsive to the first
value corresponding to the combination of values, indicating a match for the
first value.

51. The system of claim 50, wherein the recursive function module:
- responsive to one of the values in the second input parameter equaling the first input parameter, returns a value list including the equal value;
 - responsive to none of the values in the second input parameter equaling the first input parameter, and the second parameter containing only one value, returns an indicator that no match was found;
 - responsive to none of the values in the second input parameter equaling the first input parameter, and the second parameter containing more than one value, performs the recursive submethod using a modified first input parameter and a modified second input parameter, the modified second input parameter omitting a selected value and the modified first input parameter being obtained by subtracting the selected value from the first input parameter.
52. The system of claim 50, wherein the recursive function module:
- responsive to one of the values in the second input parameter equaling the first input parameter, returns a value list including the equal value;
 - responsive to none of the values in the second input parameter equaling the first input parameter, and the second parameter containing only one value, returns an indicator that no match was found;
 - responsive to none of the values in the second input parameter equaling the first input parameter, and the second parameter containing more than one value, performs the steps of:
 - a) selecting a value in the second input parameter;

- b) subtracting the selected value from the first input parameter to obtain a modified first input parameter;
- c) generating a modified value list including all values in the second input parameter except the selected value;
- d) performing the recursive submethod using a first input parameter including the modified first input parameter and a second input parameter including the modified value list;
- e) responsive to the recursive submethod returning a value list, adding the selected value to the returned list to generate a modified value list, and returning the modified value list;
- f) responsive to the recursive submethod returning an indicator that no match was found, performing the steps of:
 - responsive to any values remaining in the second input parameter, repeating steps a) through f); and
 - responsive to no values remaining in the second input parameter, returning an indicator that no match was found.

53. The system of claim 50, wherein each value is associated with a transaction.

54. A system for matching a first combination of at least two values with a second combination of at least two values in a list of values, the system comprising:

- a first input device, for obtaining each value in the first combination;
- coupled to the first input device, a combination module, for combining the obtained values to generate a first combined value;
- a second input device, for obtaining the second list of values;

coupled to the combination module and the second input devices, a memory for storing the first value and the second list;

coupled to the memory, a recursive function module, for performing a recursive function, using a first input parameter including the first combined value and a second input parameter including the second list of values, to determine whether the first combined value corresponds to a second combination of values from the second list; and

coupled to the recursive function module, a match indicator for, responsive to the first combined value corresponding to the second combination of values, indicating a match for each value in the first combination.

Evidence Appendix

None.

Related Proceedings Appendix

None.